WHAT IS CLAIMED IS:

1. A system for validating an interface of a dynamically linkable component, comprising:

a check code generator that transforms said interface of said dynamically linkable component into an interface identifier representing said interface and couples said interface identifier to said dynamically linkable component; and

a interface verifier that employs said interface identifier to determine a compatibility of said dynamically linkable component.

2. The system as recited in Claim 1 wherein said check code generator transforms said interface of said dynamically linkable component into said interface identifier by transforming a textual representation of at least a portion of said interface.

3. The system as recited in Claim 1 wherein said check code generator couples said interface identifier to said dynamically linkable component by placing said interface identifier in a types declaration file.

4. The system as recited in Claim 1 wherein said interface identifier varies as a function of a version of said dynamically linkable component.

5. The system as recited in Claim 1 wherein said interface verifier employs said interface identifier to determine a compatibility of said dynamically linkable component with a second dynamically linkable component.

6. The system as recited in Claim 1 wherein said interface verifier is a part of a second dynamically linkable component.

7. The system as recited in Claim 1 wherein said interface verifier determines a compatibility of said dynamically linkable component by comparing said interface identifier with a history list containing at least one member.

8. The system as recited in Claim 1 wherein said interface identifier is a type selected from the group consisting of:

a check sum, and

a cyclic redundancy check.

9. The system as recited in Claim 1 wherein said check code generator uses filtering directives to include and exclude portions of said interface from said interface identifier.

10. A method of validating an interface of a dynamically linkable component, comprising:

transforming said interface of said dynamically linkable component into an interface identifier representing said interface;

coupling said interface identifier to said dynamically linkable component; and

employing said interface identifier to determine a compatibility of said dynamically linkable component.

11. The method as recited in Claim 10 wherein said transforming comprises transforming a textual representation of at least a portion of said interface.

12. The method as recited in Claim 10 wherein said coupling comprises placing said interface identifier in a types declaration file.

13. The method as recited in Claim 10 wherein said interface identifier varies as a function of a version of said dynamically linkable component.

14. The method as recited in Claim 10 wherein said employing comprises employing said interface identifier to determine a compatibility of said dynamically linkable component with a second dynamically linkable component.

15. The method as recited in Claim 10 wherein said interface verifier is a part of a second dynamically linkable component.

16. The method as recited in Claim 10 wherein said employing comprises comparing said interface identifier with a history list containing at least one member.

17. The method as recited in Claim 10 wherein said interface identifier is a type selected from the group consisting of:

a check sum, and

a cyclic redundancy check.

18. The method as recited in Claim 10 wherein said transforming uses filtering directives to include and exclude portions of said interface from said interface identifier.

Sub a7

~~19. A system for validating an interface of a dynamically linkable component, comprising:~~

~~an interface identifier, coupled to said dynamically linkable component, that represents said interface of said dynamically linkable component; and~~

~~a interface verifier that employs said interface identifier to determine a compatibility of said dynamically linkable component.~~

20. The system as recited in Claim 19 wherein said interface identifier is contained within a types declaration file.

21. The system as recited in Claim 19 wherein said interface identifier varies as a function of a version of said dynamically linkable component.

22. The system as recited in Claim 19 wherein said interface verifier employs said interface identifier to determine a compatibility of said dynamically linkable component with a second dynamically linkable component.

23. The system as recited in Claim 19 wherein said interface verifier is a part of a second dynamically linkable component.

24. The system as recited in Claim 19 wherein said interface verifier determines a compatibility of said dynamically linkable component by comparing said interface identifier with a history list containing at least one member.

Sub B

~~25. A method of validating an interface of a dynamically linkable component, comprising:~~

~~coupling an interface identifier to said dynamically linkable component; and~~

~~employing said interface identifier to determine a compatibility of said dynamically linkable component.~~

26. The method as recited in Claim 25 wherein said coupling comprises placing said interface identifier in a types declaration file.

27. The method as recited in Claim 25 wherein said interface identifier varies as a function of a version of said dynamically linkable component.

28. The method as recited in Claim 25 wherein said employing comprises employing said interface identifier to determine a compatibility of said dynamically linkable component with a second dynamically linkable component.

29. The method as recited in Claim 25 wherein said interface verifier is a part of a second dynamically linkable component.

30. The method as recited in Claim 25 wherein said employing comprises comparing said interface identifier with a history list containing at least one member.

Sub a4

31. A real-time process control system, comprising:

a plurality of sensors and controllable devices;

a controller, coupled to said plurality of sensors and controllable devices, that executes software having at least first and second dynamically linkable components to coordinate an operation of said plurality of sensors and controllable devices;

an interface identifier, coupled to said first dynamically linkable component, that represents an interface of said first dynamically linkable component; and

a interface verifier that employs said interface identifier to determine a compatibility of said first and second dynamically linkable components.

32. The real-time process control system as recited in Claim 31 wherein said interface identifier is a transformation of a textual representation of at least a portion of said interface.

33. The real-time process control system as recited in Claim 31 wherein said interface identifier is contained within in a types declaration file.

34. The real-time process control system as recited in Claim 31 wherein said interface identifier varies as a function of a version of said first dynamically linkable component.

35. The real-time process control system as recited in Claim 31 wherein said interface verifier is a part of said second dynamically linkable component.

36. The real-time process control system as recited in Claim 31 wherein said interface verifier determines a compatibility of said first dynamically linkable component by comparing said interface identifier with a history list associated with said second dynamically linkable component and containing at least one member.

37. The real-time process control system as recited in Claim 31 wherein said interface identifier is a type selected from the group consisting of:

a check sum, and

a cyclic redundancy check.